**Coding Activity 3.2**

**Build An HTML Form**

**Submitting A Form For Pizza**

Now that you've seen some examples of forms, it's time to get familiar with the types of fields that can be present on forms. It's important to know the best features of the tool you are using to implement it most effectively.

**Your Pizza Form**

Your form needs to gather the following information:

* Size
* Pepperoni
* Gluten-Free
* Name for the order
* Quantity
* Additional Instructions

**Building blocks**

Inside HTML forms, you can use a variety of fields. Provided below are examples of the fields you can use in your pizza form:

First is the <input> tag. Examples of input tags are shown below:

A checkbox that holds a checked field of true or false to indicate whether the box is checked or not.

<input id="example-checkbox-input" type="checkbox"/>

A number that allows you to increment and decrement the number value stored in the field with built-in buttons.

<input type="number" id="example-number-input"/>

A text field that allows you to enter a string

<input id="example-text-input" type="text"/>

Next is the <select> tag, which allows the selection of <option> child elements through a UI dropdown. Here is an example:

<select id="example-select">

<option>Freshman</option>

<option>Sophomore</option>

<option>Junior</option>

<option>Senior</option>

</select>

The last tag you will need for your pizza form is the <textarea> tag. This one allows you to define a box similar to the input field with type text except that it's designed for larger inputs (paragraph size). It allows you to specify attributes such as cols, which is the number of characters lengthwise and rows, which is the number of lines heightwise. The example is shown below: <textarea id="example-textarea" cols="50" rows="3"/>

**Instructions**

You need to add the following form fields to the existing form returned by the render function within the form.jsx file.

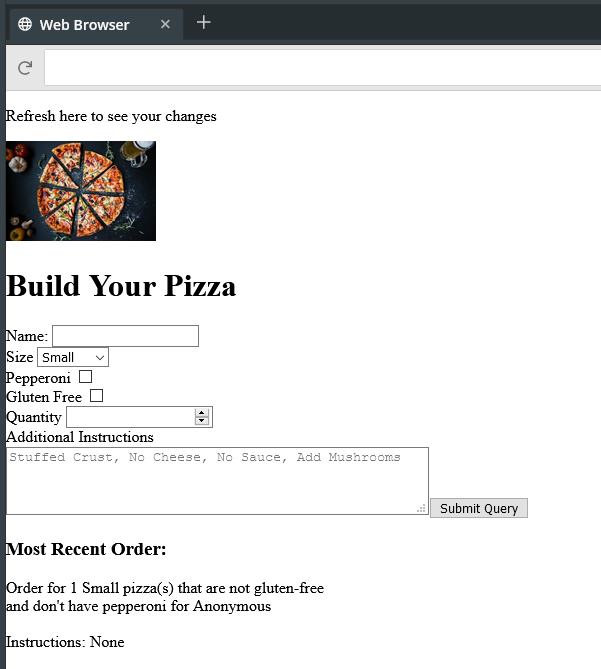
1. First gather the name for the order. This field should include: an <input> tag of type text and have an id of "name-for-order"
2. Next is the size field, which should be a <select> tag, have options for small, medium, and large, and have an id of "size"
3. Now, a checkbox for pepperoni, which should be an <input> tag with checkbox type and have an id of "pepp"
4. Another checkbox for gluten-free, which should be an <input> tag with checkbox type and have an id of "g-free"
5. Next, a number field for quantity, which should be an <input> tag with number type and have an id of "quantity"
6. Finally, a text area for any additional instructions such as "extra cheese" or "stuffed crust", which should be a <textarea> tag and have a cols attribute value of 50, have a rows attribute value of 3, and an id of "add-instr"

Using the examples above, you should be able to create all of these fields to make your very own pizza form. See the picture at the bottom of the instructions section for an example of how the completed product should look with an order submitted.

*Hints:*

* *To see more examples, checkout the*[*official documentation for form inputs*](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input)
* *Note the element ID's referenced in the handleSubmit function. It is imperative that you match those IDs on the elements you create according to instructions for this activity to work correctly*

Example of how the form should look:

[](https://cdn.filestackcontent.com/OJYyNpHQfC8sKYMY4sXm)

Tasks

Add the fields for size, pepperoni, and gluten free in the form.jsx file. Make sure the id's match what the handleSubmit function expects.

**3**

Add the fields for name, quantity, and additional instructions to the provided form in the form.jsx file. Make sure the id's match what the handleSubmit function expects.

**3**

All fields are implemented correctly so that on submission of the Build Your Pizza form, a valid order string is constructed, alerted, and stored in the most recent order div.